DNS, DoT and DoH
DoQ Not included

Alan Clegg
December 11, 2019

RFC 1034 (DNS)
RFC 7858 (DoT)
RFC 8484 (DoH)
draft-huitema-quic-dnsoquic-06 (DoQ)
Agenda

• How this presentation is different
• Differentiate the available technologies
• Observe the current environment

• Consider an actual implementation
• Thoughts on debugging
• Future conversations
• Closing comments
Rationale

The right answer is that everyone should be running a feature-complete caching + forwarding resolver on localhost. All the rest of these discussions are noise from companies that want eyeballs. twitter.com/CarolineGreer/...

Phillip Remaker @philrem
@davifu The target audience for DoH doesn’t know any of those words.

via Twitter for iPhone in reply to @davifu
What are we talking about?
• What we know as “DNS” today

• TCP/UDP port 53

• Not encrypted
  Easily monitored
  Easily blocked
  Easily redirected
  Easily modified (unless DNSSEC implemented)
DoT

- DNS over TLS
  - Uses same encryption as HTTPS
- TCP port 853
- Encrypted
  - Easily monitored (for traffic, not content)
  - Easily blocked
  - Not easily modified
  - More CPU intensive (TLS setup per server contacted)
DoH

- DNS over HTTPS
- TCP port 443
- Encrypted
  Not easily monitored (mixed in HTTPS traffic)
  Not easily blocked
  Not easily redirected
  Not easily modified
## Enter the Matrix

<table>
<thead>
<tr>
<th>Can it Be…</th>
<th>Do53</th>
<th>DoT</th>
<th>DoH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>Yes, Plain Text</td>
<td>No, Encrypted</td>
<td>No, Encrypted</td>
</tr>
<tr>
<td>Monitored</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>(Traffic, not Content)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Blocked</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Modified</td>
<td>Yes, Unless using DNSSEC</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Redirected</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
What does this look like?
Setup

Home/Office User
ISP DNS Servers
The Watcher
Alternative (Cloud) DNS
The Watcher

ISP DNS Servers

Home/Office User

The Watcher

Alternative (Cloud) DNS
DoT
DoH
ISP DNS Servers

Alternative (Cloud) DNS

The Watcher

DoH

Attack of The Apps
DoH

Attack of The Malware

Home/Office User

ISP DNS Servers

The Watcher

Alternative (Cloud) DNS
Imagine an image of a yellow man with MG hair/ears saying "DoH!"
Where are we now?
DoT

- There are implementations and deployments
  - Client support in Android 9+
  - Server very easy to configure as an nginx stream
    - [https://www.aaflalo.me/2019/03/dns-over-tls/](https://www.aaflalo.me/2019/03/dns-over-tls/)
  - Supported in dnsdist
DoH

• List of known DoH servers

https://github.com/curl/curl/wiki/DNS-over-HTTPS#publicly-available-servers

• The problem with firewalling DoH (blocking port 443) is that if the remote server also serves web content, it is impossible to block without losing access to the content.

• What if www.google.com also responded to DoH queries?
DoH

- Google
- Cloudflare
- Quad9
- CleanBrowsing
- @chantra
- @jedisct1
- PowerDNS
- blahdns.com
- NekomimiRouter.com
- SecureDNS.eu
- Rubyfish.cn
- Commons Host
- dnswarden.com
- aaflalo.me
- Server EU
- Foundation for Applied Privacy
DoH

- Google
- Cloudflare
- Quad9
- CleanBrowsing
- @chantra
- @jedisct1
- PowerDNS
- blahdns.com
- NekomimiRouter.com
- SecureDNS.eu
- Rubyfish.cn
- Commons Host
- dnswarden.com
- aaflalo.me
- Server EU
- Foundation for Applied Privacy

AND ALL THE OTHERS
DoH

### Supported in browsers and clients

<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firefox</td>
<td>62</td>
<td>temporary docs</td>
</tr>
<tr>
<td>Bromite</td>
<td>67.0.3396.88</td>
<td>How to enable DoH</td>
</tr>
<tr>
<td>curl</td>
<td>7.62.0</td>
<td>See DOH-implementation</td>
</tr>
<tr>
<td>OkHttp</td>
<td>3.11</td>
<td>See Providers</td>
</tr>
<tr>
<td>curl-doh</td>
<td>n/a</td>
<td>basic stand-alone DoH client that uses curl</td>
</tr>
</tbody>
</table>

And here's news...

- Microsoft has announced the addition of DoH to Windows

Real Life Example
Implementation

- Branch Office:
  - Star Brilliant's "High performance DNS over HTTPS client & server" as client
    - Listens on ethernet interface, port 53
    - Internal, no protective ACLs
    - Sends "internal" qnames to localhost:53
  - Others: https://doh.clegg.com/dns-query
Implementation

- Branch Office:
- BIND 9.14.7
  - Listens on loopback interface, port 53
  - Private TLD + in-addr.arpa
Implementation

• Cloud 1:
  • Linode - Debian 9
  • Star Brilliant's "High performance DNS over HTTPS client & server" - as Server
  • BIND 9.15.6 (development branch)
  • Nginx 1.16.1
Implementation

• Cloud 1 Configuration:
  • BIND listens on all interfaces (IPv4 & IPv6)
    • Port 53 / Recursive
    • ACLs allow queries from localhost and Cloud 2 only
    • Blacklisting / Adblocking via custom scripts
Implementation

• Cloud 1 Configuration:

  • Nginx

    • Existing install

    • Stream accepting DoT connections on 853

    • Reverse proxy for http://<name>/dns-query/

    • Feeds query to http://localhost:8053
Implementation

• Cloud 1 Configuration:

  • `doh-server` (Star Brilliant)
    • Listening on `localhost:8053`
    • Accepts DoH queries
    • Converts raw query to DNS query
    • Passes them to `localhost:53`
Implementation

• Cloud 2:
  • Linode - Debian 10
  • dnsdist (from git)
  • BIND 9.14.8
  • Nginx
Implementation

• Cloud 2 Configuration:
  • BIND
    • Listening on localhost (v4 & v6) port 5353
  • Recursive
Implementation

- Cloud 2 Configuration:
  - dnsdist
  - Listening externally on 853 (DoT) & 443 (DoH)
  - Load balances queries to:
    - localhost : 5353
    - Cloud 1 : 53
Debugging
Debugging this mess

- Fantastic external resources:
  - https://getdnsapi.net/query/
    - Allows testing of DNS over varying combinations of transport (UDP, TCP, TLS)
  - https://github.com/dcidd/
    - DoT and DoH command line (PHP) clients
Debugging this mess

Dec 10 15:45:01 stargate doh-client[27418]: 2019/12/10 15:45:01 Request "ptz-cam.boat. IN A" is passed through 127.0.0.2:53.
Dec 10 15:45:01 stargate doh-client[27418]: 2019/12/10 15:45:01 choose upstream: upstream type: IETF, upstream url: https://doh.clegg.com/dns-query
Dec 10 15:45:01 stargate doh-client[27418]: 2019/12/10 15:45:01 choose upstream: upstream type: IETF, upstream url: https://doh.clegg.com/dns-query


• Turn up logging
  • You can't see much in packet dumps
  • Oh ... privacy? Yeah, about that...

• Software is young
  • Log messages from same daemon with same data in different columns
  • Able to change without breaking everyone!
Debugging this mess

aclegg@stargate:~/.local/bin $ doh-client \
>     --domain dns.dnsoverhttps.net \
>     --qname sigfail.verteiltesysteme.net \
>     --dnssec 
flag provided but not defined: --domain
Usage of doh-client:
  -conf string
    Configuration file (default "doh-client.conf")
  -pid-file string
    PID file for legacy supervision systems lacking support for reliable cgroup-based process tracking
  -verbose
    Enable logging
  -version
    Show software version and exit
aclegg@stargate:~/.local/bin $ ./doh-client --domain dns.dnsoverhttps.net \
>     --qname sigfail.verteiltesysteme.net --dnssec 
Traceback (most recent call last):
  File "/home/aclegg/.local/lib/python3.7/site-packages/aioh2/__init__.py", line 2, in <module>
      from .helper import *
    File "/home/aclegg/.local/lib/python3.7/site-packages/aioh2/helper.py", line 89
      async_task = asyncio.async
          ^
SyntaxError: invalid syntax

• Everything is named the same
  • And it explodes...
Debugging this mess

aclegg@stargate:~$ cat getchain
echo | openssl s_client -connect $1:853 |grep -B 2 -A 5 "Certificate chain"

aclegg@stargate:~$ getchain dns.google
depth=2 OU = GlobalSign Root CA - R2, O = GlobalSign, CN = GlobalSign
verify return:1
depth=1 C = US, O = Google Trust Services, CN = GTS CA 1O1
verify return:1
depth=0 C = US, ST = California, L = Mountain View, O = Google LLC, CN = dns.google
verify return:1
DONE
CONNECTED(00000003)
---
Certificate chain
  0 s:C = US, ST = California, L = Mountain View, O = Google LLC, CN = dns.google
  i:C = US, O = Google Trust Services, CN = GTS CA 1O1
  1 s:C = US, O = Google Trust Services, CN = GTS CA 1O1
  i:OU = GlobalSign Root CA - R2, O = GlobalSign, CN = GlobalSign
---

• Learn about OpenSSL!
  • Paid certificates from CA
  • Free certificates from Let's Encrypt
  • No more self-signed certificates if possible
Debugging this mess

```
aclegg@stargate:~ $ dnstls alan.clegg.com doh.clegg.com
alan.clegg.com has address 45.33.100.174
aclegg@stargate:~ $ dnstls some-bad-site.com doh.clegg.com
pornhub.com has address 66.254.114.41
aclegg@stargate:~ $ dnstls some-bad-site.com cleanbrowsing
Host pornhub.com not found: 3(NXDOMAIN)
aclegg@stargate:~ $ dnstls 00author.com doh.clegg.com
00author.com has address 0.0.0.0
aclegg@stargate:~ $ dnstls 00author.com cleanbrowsing
00author.com has address 64.136.20.41
```

• Everything that looks simple ... Isn't.
  • Wrapping DNS queries in TLS
  • Generating a DNS query over HTTPS

• Which client is talking to which server?
  • Things blocked (firewalled) on one server are not blocked on another
  • Blocking methods differ so results will differ
  • Applications on the same client may be talking to different servers with different policies

• Caches are now all over the place
  • In the DoH/DoT code, in the recursive server, in the client…
Debugging this mess

• Learn your data paths
  • No longer directly from client to resolver to auth
  • Where did that query go / vanish?

• Keep software up-to-date
  • Rapidly changing
  • Not yet packaged - dependency hell
- Learn your data paths
- No-longer directly from client to resolver to auth
- Where did that query go / vanish?
Conversations We Are Going To Have
Recent Conversation
Starters

David Ulevitch @davidu
6 Dec
The right answer is that everyone should be running a feature-complete caching + forwarding resolver on localhost. All the rest of these discussions are noise from companies that want eyeballs. twitter.com/CarolineGreer/...
via Twitter Web App

Bert Hubert @PowerDNS_Bert
7 Dec
@davidu @jpmens I struggle with this - what should they forward to? A single-user resolver has terrible performance, but sending your traffic some a random cache also has downsides. I'd love to somehow square the privacy, performance & reliability circle.
via Twitter Web App in reply to @davidu
Bert Hubert @PowerDNS_Bert

"We must do DNS over HTTPS because DNS is unencrypted"

Large provider listens and provides a public DoH service

"No not like that".

Jan-Piet MENS @jpmens

@PowerDNS_Bert I thought DoH was meant to get encrypted DNS so that the "spooky ISP" wouldn’t have access to queries. Now I read “Naturally ISPs that don’t need to filter, manipulate or snoop on DNS traffic will find it much easier to establish their own DoH solution” and I am shocked. not.
Recent Conversation
Starters

Paul_IPv6 @Paul_IPv6
@kolkman i am encouraged that several very large consumer ISPs are already doing trials of DoT and DoH, so we should have a better idea of at least what more TCP might cost in terms of hardware/network support. and i would still love to see stub DNSSEC validating in end devices as a start
by Olaf Kolkman via Twitter for iPhone

Paul_IPv6 @Paul_IPv6
@kolkman another piece we need to consider/test is that current DoT/DoH is designed for last mile, not to auth. ADoT is going to need different tuning and experience. when do persistant connections, multiple response per connection, etc make sense operationally? lots to research yet.
by Olaf Kolkman via Twitter for iPhone
Moving forward..

- We have the technology available to deploy
  - Will it scale?
  - Is it supportable?
    - My stuff broke, who's going to fix it?
  - Timeframe for support in BIND
Software Links

• BIND installed from source code (ftp.isc.org)

• https://github.com/m13253/dns-over-https

• https://github.com/PowerDNS/pdns
Contact Information

• Alan Clegg
  • E-Mail: aclegg@isc.org
  • Twitter: @AlanAtISC
Questions?

Comments?
https://www.isc.org